**Web-based Ordering &Ingredient Estimating for Bakery Manufacturer**

Software Test Record

By

**Miss. Nontra Nonsee 542115026**

**Mr. Parinya Panyanak 542115034**

Department of Software Engineering

College of Arts, Media and Technology

Chiang Mai University

Project Advisor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Mr.Phudinan Singkhamfu**

**Document History**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Document Name** | **Version** | **Status** | **Date** | **Viewable** | **Reviewer** | **Responsible** |
| **Documents** | | | | | | |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer– Test Record\_V.1.5.docx | V.1.5  - Add Chapter1  - Add Chapter 2  - Add Chapter 3  - Fix inappropriate words | Drafted | 19-03-2015 | NN, PP, PS | NN, PP | NN, PP |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Record \_V.1.7.docx | V.1.7  - Adjust Chapter 2  - Adjust Chapter 3  - Fix inappropriate words | Drafted | 21-03-2015 | NN, PP, PS | NN, PP | NN, PP |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Record \_V.2.0.docx | V.2.0  - Adjust Chapter 2  - Adjust Acronyms & Definitions | Drafted | 22-03-2015 | NN, PP, PS | NN, PP | NN, PP |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer – Test Record \_V.3.0.docx | V.3.0  - Adjust Chapter 1  - Adjust Chapter 2  - Adjust Chapter 3  - Adjust Acronyms & Definitions  - Fix inappropriate words | Released | 30-06-2015 | NN, PP, PS | NN, PP | NN, PP |

**\*NN = Nontra Nonsee**

**\*PP = Parinya Panyanak**

**\*PS = Phudinan Singkhamfu**

**Table of contents**

[**Chapter One | Introduction** 4](#_Toc423422464)

[1.1 Purpose 4](#_Toc423422465)

[1.2 Scope 4](#_Toc423422466)

[1.3 Responsibility 4](#_Toc423422467)

[1.4 Acronyms and Definitions 4](#_Toc423422468)

[**Chapter Two | Unit Testing Record** 5](#_Toc423422469)

[**Chapter Three | System Testing Record** 19](#_Toc423422470)

# **Chapter One | Introduction**

## 1.1 Purpose

The purpose of Software Test Record Document describes the test record for the Web-based Ordering & Ingredient Estimating for Bakery Manufacturer System. This document is recording the actual result and pass/fail criteria of each test case that have design in test plan and execute. In this document, provides test record of all project progress report. The document consists of member management, product management, order management, ingredient management, ingredient estimation, and report system.

## 1.2 Scope

This Software Test Record Document describes the test record for the Web-based Ordering & Ingredient Estimating for Bakery Manufacturer System.

## 1.3 Responsibility

|  |  |  |
| --- | --- | --- |
| **Test Process** | | |
| **Test** | **Verification** | **Responsibility** |
| Record unit testing of web application | All members  Advisor | ParinyaPanyanak |
| Record system testing of web application |

## 1.4 Acronyms and Definitions

Pass Actual Result is follow Expect output

Fail Actual Result is not follow Expect output

UTR Unit Test Record

STR System Test Record

UI User Interface

# **Chapter Two | Unit Testing Record**

**UTR-01:** public ActionResultLogin (string returnUrl)

**Test date: 1**4 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Login", "\_PlainLayout"); | return View ("Login", "\_PlainLayout"); | Pass |

**UTR-02:** public ActionResultLogin (LoginModel model, string returnUrl)

**Test date: 1**4 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | RedirectToLocal (null); | RedirectToLocal (null); | Pass |
| 2 | return View ("Login", "\_PlainLayout",  { Username == “[tung@dd.com](mailto:tung@dd.com)”,  Password == “555555”,  RememberMe == false}); | return View ("Login", "\_PlainLayout",  { Username == “[tung@dd.com](mailto:tung@dd.com)”,  Password == “555555”,  RememberMe == false}); | Pass |

**UTR-03:** public ActionResultIndex ()

**Test date: 1**4 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Index", List<Member>); | return View ("Index", List<Member>); | Pass |

**UTR-04:** public ActionResultDetails (int id)

**Test date: 1**4 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | Return View ("Details",  {Username == “[tung@gmail.com](mailto:tung@gmail.com)”,  Password == “123456”,  firstName == “Parinya” ,  lastName == “Panyanak”,  phoneNumber == “0863338313”,  address == “xxxxxx”}); | Return View ("Details",  {Username == “[tung@gmail.com](mailto:tung@gmail.com)”,  Password == “123456”,  firstName == “Parinya” ,  lastName == “Panyanak”,  phoneNumber == “0863338313”,  address == “xxxxxx”}); | Pass |
| 2 | return HttpNotFound (); | return HttpNotFound (); | Pass |

**UTR-05:** public ActionResultUserProfile (string id)

**Test date: 1**6 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("UserProfile",  {Username == “[tung@dd.com](mailto:tung@dd.com)”,  Password == “123456”, firstName == “tung”,  lastName == “lastname”,  phoneNumber == “089133423”,  address == “xxxxxx”});); | return View ("UserProfile",  {Username == “[tung@dd.com](mailto:tung@dd.com)”,  Password == “123456”, firstName == “tung”,  lastName == “lastname”,  phoneNumber == “089133423”,  address == “xxxxxx”});); | Pass |
| 2 | return HttpNotFound (); | return HttpNotFound (); | Pass |

**UTR-06:** public ActionResultCreate ()

**Test date: 1**6 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Create"); | return View ("Create"); | Pass |

**UTR-07:** public ActionResultCreate (Member member)

**Test date: 1**6 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return RedirectToAction ("Details",  { Username == [nontra@gmail.com](mailto:nontra@gmail.com) ,  Password == nontra789,  firstName == Nontra,  lastName == Mahachanont,  phoneNumber == 0875885737,  address == 42/1 m.6 Muang Chaingmai}); | return RedirectToAction ("Details",  { Username == [nontra@gmail.com](mailto:nontra@gmail.com) ,  Password == nontra789,  firstName == Nontra,  lastName == Mahachanont,  phoneNumber == 0875885737,  address == 42/1 m.6 Muang Chaingmai}); | Pass |
| 2 | return View (); | return View (); | Pass |

**UTR-08:** public ActionResultEdit (int id = 0)

**Test date: 1**7 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Edit",  {Username == “[tung@gmail.com](mailto:tung@gmail.com)”,  Password == “123456”,  firstName == “Parinya”,  lastName == “Panyanak”,  phoneNumber == “0863338313”,  address == “xxxxxx”});); | return View ("Edit",  {Username == “[tung@gmail.com](mailto:tung@gmail.com)”,  Password == “123456”,  firstName == “Parinya”,  lastName == “Panyanak”,  phoneNumber == “0863338313”,  address == “xxxxxx”});); | Pass |
| 2 | return HttpNotFound (); | return HttpNotFound (); | Pass |

**UTR-09:** public ActionResultEdit (Member member)

**Test date: 1**7 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Index") | return View ("Index") | Pass |
| 2 | return View ({Username == “[tortung](mailto:tung@gmail.com)”,  Password == “123456”,  firstName == “Parinya”,  lastName == “Panyanak”,  phoneNumber == “0863338313”,  address == “xxxxxx”});); | return View ({Username == “[tortung](mailto:tung@gmail.com)”,  Password == “123456”,  firstName == “Parinya”,  lastName == “Panyanak”,  phoneNumber == “0863338313”,  address == “xxxxxx”});); | Pass |

**UTR-10:** public ActionResultDelete (int id = 0)

**Test date: 1**7 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Delete",  {Username == “[tung@gmail.com](mailto:tung@gmail.com)”, Password == “123456”,  firstName == “Parinya” ,  lastName == “Panyanak”,  phoneNumber == “0863338313”, address == “xxxxxx”}); | return View ("Delete",  {Username == “[tung@gmail.com](mailto:tung@gmail.com)”, Password == “123456”,  firstName == “Parinya” ,  lastName == “Panyanak”,  phoneNumber == “0863338313”, address == “xxxxxx”}); | Pass |
| 2 | return HttpNotFound (); | return HttpNotFound (); | Pass |

**UTR-11:** public ActionResultDeleteConfirmed (int id)

**Test date: 1**7 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Index"); | return View ("Index"); | Pass |
| 2 | return View (user); | return View (user); | Pass |

**UTR-12:** public ActionResultIndex()

**Test date: 1**7 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | Return View ("Index", List<Product>) | Return View ("Index", List<Product>) | Pass |

**UTR-13:** public ActionResultDetails (int id = 0)

**Test date: 1**7 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Details", {Name == “Longgang cake”, Price == 50, Category == cake,  Picture == \\UploadImages\\2images (4).jpg}); | return View ("Details", {Name == “Longgang cake”, Price == 50, Category == cake,  Picture == \\UploadImages\\2images (4).jpg}); | Pass |
| 2 | return HttpNotFound (); | return HttpNotFound (); | Pass |

**UTR-14:** public ActionResultCreate ()

**Test date: 1**7 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Create"); | return View ("Create"); | Pass |

**UTR-15:** public ActionResultCreate(Product product)

**Test date: 1**7 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return RedirectToAction ("Index"); | return RedirectToAction ("Index"); | Pass |
| 2 | return View ({Name == “banana tart”,  Price == zxc, Category == “tart”  Picture == \UploadImages\5ดาวน์โหลด.jpg}); | return View ({Name == “banana tart”,  Price == zxc, Category == “tart”  Picture == \UploadImages\5ดาวน์โหลด.jpg}); | Pass |

**UTR-16:** public ActionResultEdit (int id = 0)

**Test date:** 24 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Edit", ",  {Name == “Longgang cake”,  Price == 50, Category == cake ,  Picture == \\UploadImages\\2images (4).jpg}); | return View ("Edit", ",  {Name == “Longgang cake”,  Price == 50, Category == cake ,  Picture == \\UploadImages\\2images (4).jpg}); | Pass |
| 2 | return HttpNotFound(); | return HttpNotFound(); | Pass |

**UTR-17:** public ActionResultEdit (Product product)

**Test date:** 24 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return RedirectToAction ("Index", "Product", {Name == “Coconut cake”,  Price == 50, Category == cake ,  Picture == \\UploadImages\\2images (4).jpg}); | return RedirectToAction ("Index", "Product", {Name == “Coconut cake”,  Price == 50, Category == cake ,  Picture == \\UploadImages\\2images (4).jpg}); | Pass |
| 2 | return View ({",{Name == “Coconut cake”, Price == xyz, Category == cake ,  Picture == \\UploadImages\\2images (4).jpg}); | return View ({",{Name == “Coconut cake”, Price == xyz, Category == cake ,  Picture == \\UploadImages\\2images (4).jpg}); | Pass |

**UTR-18:** public ActionResultDelete (int id = 0)

**Test date:** 24 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Delete", {Name == “Longgang cake”, Price == 50, Category == cake ,  Picture == \\UploadImages\\2images (4).jpg}); | return View ("Delete", {Name == “Longgang cake”, Price == 50, Category == cake ,  Picture == \\UploadImages\\2images (4).jpg}); | Pass |
| 2 | return HttpNotFound(); | return HttpNotFound(); | Pass |

**UTR-19:** public ActionResultDeleteConfirmed (int id)

**Test date:** 24 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View("Index"); | return View("Index"); | Pass |
| 2 | return View (product); | return View (product); | Pass |

**UTR-20:** public ActionResultLogOut ()

**Test date:** 24 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return RedirectToAction ("Index", "Home"); | return RedirectToAction ("Index", "Home"); | Pass |

**UTR-21:** private ActionResultRedirectToLocal (string returnUrl)

**Test date:** 24 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return RedirectToAction ("Index", "Home"); | return RedirectToAction ("Index", "Home"); | Pass |
| 2 | return Redirect (returnUrl); | return Redirect (returnUrl); | Pass |

**Unit Test Case-22 (UTC-22):** public ActionResultDetails (int id = 0)

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Details", {Name == “Raisin”,  Size == 50, ingredID ==2,  Category == Raisin, Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg}); | return View ("Details", {Name == “Raisin”,  Size == 50, ingredID == 2,  Category == Raisin, Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg}); | Pass |
| 2 | return HttpNotFound (); | return HttpNotFound (); | Pass |

**Unit Test Case-23 (UTC-23):** public ActionResult Create ()

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Create"); | return View ("Create"); | Pass |

**Unit Test Case-24 (UTC-24):** public ActionResultCreate (Ingredient ingred)

**Test date:** 24 April 2015**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return RedirectToAction ("Index"); | return RedirectToAction ("Index"); | Pass |
| 2 | {Name ==“Raisin”, Size == xxx,  Category == Raisin , Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg } | {Name ==“Raisin”, Size == xxx,  Category == Raisin , Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg } | Pass |

**Unit Test Case-25 (UTC-25):** public ActionResultEdit (int id = 0)

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Edit",{Name == “Raisin”,  Size == 1, ingredID ==2, Category == Raisin, Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg}); | return View ("Edit",{Name == “Raisin”,  Size == 1, ingredID ==2, Category == Raisin, Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg}); | Pass |
| 2 | return HttpNotFound (); | return HttpNotFound (); | Pass |

**Unit Test Case-26 (UTC-26):** public ActionResultEdit (Ingredient ingre)

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return RedirectToAction ("Index", "Ingredient", {Name ==“Raisin”, Size == 20, Category == Raisin, Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg } | return RedirectToAction ("Index", "Ingredient", {Name ==“Raisin”, Size == 20, Category == Raisin, Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg } | Pass |
| 2 | return View {Name ==“Raisin”, Size == xxx, Category == Raisin , Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg }); | return View {Name ==“Raisin”, Size == xxx, Category == Raisin , Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg }); | Pass |

**Unit Test Case-27 (UTC-27):** public ActionResultDelete (int id = 0)

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Delete", {Name ==“Raisin”,  Size == 1, Category == Raisin,  Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg }); | return View ("Delete", {Name ==“Raisin”,  Size == 1, Category == Raisin,  Picture == \\UploadImages\\210733446\_10202588282381156\_594892420\_n.jpg }); | Pass |
| 2 | return HttpNotFound (); | return HttpNotFound (); | Pass |

**Unit Test Case-28 (UTC-28):** public ActionResultDeleteConfirmed (int id)

**Test date:** 29 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Index"); | return View ("Index"); | Pass |
| 2 | return View (product); | return View (product); | Pass |

**Unit Test Case-29(UTC-29):** public SendCart (Order ord, string submit)

**Test date:** 29 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("History",{id == 22, orderDate, Duedate == {10/28/2014 12:00:00 AM}, orderStatus == 0, memberId == 5}); | return View ("History",{id == 22, orderDate, Duedate == {10/28/2014 12:00:00 AM}, orderStatus == 0, memberId == 5}); | Pass |
| 2 | return View ("Product", "Home"); | return View ("Product", "Home"); | Pass |

**Unit Test Case-30(UTC-30):** public ActionResult Product ()

**Test date:** 29 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Product", List of product); | return View ("Product", List of product); | Pass |

**Unit Test Case-31(UTC-31):** public ActionResult Product (IList<Product> products)

**Test date:** 29 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | Return RedirectToAction ("Cart", "Home"); | Return RedirectToAction ("Cart", "Home"); | Pass |

**Unit Test Case-32(UTC-32):** public ActionResult History ()

**Test date:** 29 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("History", {List of wholesaler order}); | return View ("History", {List of wholesaler order}); | Pass |

**Unit Test Case-33(UTC-33):** public ActionResult Index ()

**Test date:** 29 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Index",List of order); | return View ("Index",List of order); | Pass |

**Unit Test Case-34(UTC-34):** public ActionResult Delete (int id=0)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("EditRecipe"); | return View ("EditRecipe"); | Pass |
| 2 | return View (recipe); | return View (recipe); | Pass |

**Unit Test Case-35(UTC-35):** public ActionResult Edit (int id=0)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("EditRecipe ", productID == 2,  productName == “Coconut Cake”, price==50, category == ”Cake”,  productPicture == “..\..\\UploadImages\2images (4).jpg”, information ==null} | return View ("EditRecipe ", productID == 2,  productName == “Coconut Cake”, price==50, category == ”Cake”,  productPicture == “..\..\\UploadImages\2images (4).jpg”, information ==null} | Pass |
| 2 | return View ("EditRecipe ", productID == 2,  productName == ”Coconut Cake”, price==50, category == ”Ca320”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information ==null} | return View ("EditRecipe ", productID == 2,  productName == ”Coconut Cake”, price==50, category == ”Ca320”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information ==null} | Pass |

**Unit Test Case-36(UTC-36):** public ActionResult Get (int id=0)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("EditRecipe ", productID == 2 ,  productName ==” Coconut Cake”, price == 50, category == ”Cake”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information == null} | return View ("EditRecipe ", productID == 2 ,  productName ==” Coconut Cake”, price == 50, category == ”Cake”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information == null} | Pass |
| 2 | Return HttpNotFound (); | Return HttpNotFound (); | Pass |

**Unit Test Case-37(UTC-37):** public ActionResult Index ()

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Index",List of recipe); | return View ("Index",List of recipe); | Pass |

**Unit Test Case-38(UTC-38):** public ActionResult Insert (int id=0)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("EditRecipe ", productID == 2,  productName == ”Coconut Cake”, price == 50, category == ”Cake”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information == null} | return View ("EditRecipe ", productID == 2,  productName == ”Coconut Cake”, price == 50, category == ”Cake”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information == null} | Pass |
| 2 | return View ("EditRecipe ", productID == 2,  productName == ”Coconut Cake”, price==jkjpo, category == ”Cake”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information == null} | return View ("EditRecipe ", productID == 2,  productName == ”Coconut Cake”, price==jkjpo, category == ”Cake”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information == null} | Pass |

**Unit Test Case-39(UTC-39):** public ActionResult List ()

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Index",List of recipe); | return View ("Index",List of recipe); | Pass |

**Unit Test Case-40(UTC-40):** public ActionResult GenerateReport (Report reportType, DateTime startPeriod, DateTime endPeriod)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Report ", report} | return View ("Report ", report} | Pass |
| 2 | return View (“ListAll “) | return View (“ListAll “) | Pass |

**Unit Test Case-41(UTC-41):** public ActionResult GenerateGraph(Product product, DateTime startPeriod, DateTime endPeriod)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Graph ", graph} | return View ("Graph ", graph} | Pass |
| 2 | return View (“ListAll “) | return View (“ListAll “) | Pass |

**Unit Test Case-42(UTC-42):** public ActionResult Get(int id=0)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Get", productID == 2,  productName == ”Coconut Cake”, price == 50,  category == ”Cake”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information == null); | return View ("Get", productID == 2,  productName == ”Coconut Cake”, price == 50,  category == ”Cake”, productPicture == ”..\..\\UploadImages\2images (4).jpg”, information == null); | Pass |

**Unit Test Case-43(UTC-43):** public ActionResult List(DateTime startPeriod, DateTime endPeriod)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("List ",List of order } | return View ("List ",List of order } | Pass |
| 2 | return HttpNotFound () | return HttpNotFound () | Pass |

**Unit Test Case-44(UTC-44):** public ActionResult ListAll()

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("Listall",List of order); | return View ("Listall",List of order); | Pass |

**Unit Test Case-45(UTC-45):** public ActionResult saleIngredientReport (string txtdstart, string txtdend)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("saleReport ", list of sale ingredient information} | return View ("saleReport ", list of sale ingredient information} | Pass |

**Unit Test Case-46(UTC-46):** public ActionResult saleProductReport (string txtdstart, string txtdend)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("saleProductReport ", list of sale product information} | return View ("saleProductReport ", list of sale product information} | Pass |

**Unit Test Case-47(UTC-47):** public ActionResult saleReport (string txtdstart, string txtdend)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("saleReport ", list of sale information} | return View ("saleReport ", list of sale information} | Pass |

**Unit Test Case-48(UTC-48):** public ActionResult saleStatIngredientReport (string YEAR, string txtdstart, string txtdend)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual output** | **Pass/Fail** |
| 1 | return View ("saleStatIngredientReport ", List of sale ingredient order per year } | return View ("saleStatIngredientReport ", List of sale ingredient order per year } | Pass |

**Unit Test Case-49(UTC-49):** public ActionResult saleStatistciReport (string YEAR)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Expected output** | **Pass/Fail** |
| 1 | return View ("saleStatistciReport ", List of sale order per year); | return View ("saleStatistciReport ", List of sale order per year); | Pass |

**Unit Test Case-50(UTC-50):** public ActionResult saleStatProductReport (string YEAR,string string txtdend)

**Test date:** 21 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Expected output** | **Pass/Failed** |
| 1 | return View ("saleStatProductReport ",List of sale product order per year); | return View ("saleStatProductReport ",List of sale product order per year); | Pass |

# **Chapter Three | System Testing Record**

**STR-01:** Admin can register to activate member

**Test date: 1**2 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Add new member information success and keep new information in the database. After that the system will redirect to member profile page. | Add new member information success and keep new information in the database. After that the system will redirect to member profile page. | Pass |
| 2 | Display error message “Username is required.” After that the system gives an admin input new information again. | Display error message “Username is required.” After that the system gives an admin input new information again. | Pass |
| 3 | Display error message “Password is required.” After that the system gives an admin input new information again. | Display error message “Password is required.” After that the system gives an admin input new information again. | Pass |
| 4 | Display error message “First name is required.” After that the system gives an admin input new information again. | Display error message “First name is required.” After that the system gives an admin input new information again. | Pass |
| 5 | Display error message “Last name is required.” After that the system gives an admin input new information again. | Display error message “Last name is required.” After that the system gives an admin input new information again. | Pass |
| 6 | Display error message “Phone number is required.” After that the system gives an admin input new information again. | Display error message “Phone number is required.” After that the system gives an admin input new information again. | Pass |
| 7 | Display error message “Address is required.” After that the system gives an admin input new information again. | Display error message “Address is required.” After that the system gives an admin input new information again. | Pass |
| 8 | Display error message “Please input with Email.” After that the system gives an admin input new information again. | Display error message “Please input with Email.” After that the system gives an admin input new information again. | Pass |

**STR-02:** Member can login to the system.

**Test date: 1**2 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | The system will redirect to the main layout page. | The system will redirect to the main layout page. | Pass |
| 2 | Display error message “The username or password are incorrect.” After that the system give a member input information again. | Display error message “The username or password are incorrect.” After that the system give a member input information again. | Pass |
| 3 | Display error message “Please input with Email.” After that the system gives a member input information again. | Display error message “Please input with Email.” After that the system gives a member input information again. | Pass |
| 4 | Display error message “Username is required.” After that the system gives a member input information again. | Display error message “Username is required.” After that the system gives a member input information again. | Pass |
| 5 | Display error message “Password is required.” After that the system gives a member input information again. | Display error message “Password is required.” After that the system gives a member input information again. | Pass |
| 6 | Login to the system success and keep a sign in date in the database. | Login to the system success and keep a sign in date in the database. | Pass |

**STR-03:** Member can logout from the system.

**Test date: 1**2 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Logout from the system successful. After that the system redirects to the home page. | Logout from the system successful. After that the system redirects to the home page. | Pass |

**STR-04:** Admin can manage member information by edit member.

**Test date: 1**2 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Update member information success and keep updated information in the database. After that the system will redirect to the member profile page. | Update member information success and keep updated information in the database. After that the system will redirect to the member profile page. | Pass |
| 2 | Display error message “Please input with Email.” After that the system gives an admin input information again. | Display error message “Please input with Email.” After that the system gives an admin input information again. | Pass |

**STR-05:** Admin can manage member information by delete member.

**Test date: 1**2 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Member information has removed and keep in the database. After that the system will redirect to the admin main layout page. | Member information has removed and keep in the database. After that the system will redirect to the admin main layout page. | Pass |

**STR-06:** Member can update own information.

**Test date: 1**3 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Update member information success and keep updated information in the database. After that the system will redirect to the member profile page. | Update member information success and keep updated information in the database. After that the system will redirect to the member profile page. | Pass |
| 2 | Display error message “Please input with Email.” After that the system gives an admin input information again. | Display error message “Please input with Email.” After that the system gives an admin input information again. | Pass |

**STR-07:** Member can view own personal information.

**Test date: 1**3 March 2015

**Test by:** Parinya Panyanak

**Member request view personal information.**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Display member information on the profile page. | Display member information on the profile page. | Pass |

**Member view personal information.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **First Name** | **Last Name** | **Username** | **Phone number** | **Address** | **User Type** | **Profile Picture** |
| **Expected Result** | Nontra | Mahachanont | nontra@gmail.com | 0812345678 | House No.197 Chroenrat Road, Watgate sub-district, Mueng Chiang Mai district, Chiang Mai province, 5000, Thailand | Admin | vcm_s_kf_m160_155x160.jpg |
| **Actual Result** | Nontra | Mahachanont | nontra@gmail.com | 0812345678 | House No.197 Chroenrat Road, Watgate sub-district, Mueng Chiang Mai district, Chiang Mai province, 5000, Thailand | Admin | vcm_s_kf_m160_155x160.jpg |
| **Pass/Fail** | Pass | | | | | | |

**STR-08:** Admin can view all member information

**Test date: 1**3 March 2015

**Test by:** Parinya Panyanak

**Admin search member information.**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Found information and display member information. | Found information and display member information. | Pass |
| 2 | Not found a member information and display the not found message. | Not found a member information and display the not found message. | Pass |
| 3 | Display all member are related to the filter keyword on the member information page. | Display all member are related to the filter keyword on the member information page. | Pass |

**Admin view member information.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Case** | **First Name** | **Last Name** | **Username** | **Phone number** | **Address** | **Role Name** | **Profile Picture** |
| **Expected Result** | Nontra | Mahachanont | nontra@gmail.com | 0812345678 | House No.197 Chroenrat Road, Watgate sub-district, Mueng Chiang Mai district, Chiang Mai province, 5000, Thailand | Admin | vcm_s_kf_m160_155x160.jpg |
| **Actual Result** | Nontra | Mahachanont | nontra@gmail.com | 0812345678 | House No.197 Chroenrat Road, Watgate sub-district, Mueng Chiang Mai district, Chiang Mai province, 5000, Thailand | Admin | vcm_s_kf_m160_155x160.jpg |
| **Pass/Fail** | Pass | | | | | | |

**STR-09:** Admin can manage all bakery product information by create new product.

**Test date: 1**3 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Add bakery product information success and keep new information in the database. After that the system will redirect to the product page. | Add bakery product information success and keep new information in the database. After that the system will redirect to the product page. | Pass |
| 2 | Display error message “Name is required.” After that the system gives an admin input new information again. | Display error message “Name is required.” After that the system gives an admin input new information again. | Pass |
| 3 | Display error message “Price is required.” After that the system gives an admin input new information again. | Display error message “Price is required.” After that the system gives an admin input new information again. | Pass |
| 4 | Display error message “Category is required.” After that the system gives an admin input new information again. | Display error message “Category is required.” After that the system gives an admin input new information again. | Pass |
| 5 | Display error message “Picture is required.” After that the system gives an admin input new information again. | Display error message “Picture is required.” After that the system gives an admin input new information again. | Pass |
| 6 | Display error message “Please input with a number.” After that the system gives an admin input new information again. | Display error message “Please input with a number.” After that the system gives an admin input new information again. | Pass |

**STR-10:** Admin can manage all bakery product information by edit product.

**Test date: 1**3 March 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Update bakery product information success and keep updated information in the database. After that the system will redirect to the product page. | Update bakery product information success and keep updated information in the database. After that the system will redirect to the product page. | Pass |
| 2 | Display error message “Please input with a number.” After that the system gives an admin input new information again. | Display error message “Please input with a number.” After that the system gives an admin input new information again. | Pass |

**STR-11:** Admin can manage all bakery product information by delete product.

**Test date:** 30 July 2014

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Bakery product information has deleted and keep in the database. After that the system will redirect to the product page. | Bakery product information has deleted and keep in the database. After that the system will redirect to the product page. | Pass |

**STR-12:** All user can view all bakery product information

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

**All user search bakery product information.**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Found information and display a bakery product information. | Found information and display a bakery product information. | Pass |
| 2 | Not found bakery product information and display the empty page. | Not found bakery product information and display the empty page. | Pass |
| 3 | Display all bakery products are related to with the filter keyword on the product page. | Display all bakery products are related to with the filter keyword on the product page. | Pass |

**All user view bakery product information.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Case** | **Product ID** | **Product Name** | **Information** | **Price** | **Category** | **Product Picture** |
| **Expected Result** | 12345 | Brownie | A chocolate brownie is a flat or [baked](http://en.wikipedia.org/wiki/Baking) square. | 29.00 Bath | Bread | Aglayan pasta.jpg |
| **Actual Result** | 12345 | Brownie | A chocolate brownie is a flat or [baked](http://en.wikipedia.org/wiki/Baking) square. | 29.00 Bath | Bread | Aglayan pasta.jpg |
| **Pass/Fail** | Pass | | | | | |

**System Test Case-13 (STC-13):** Wholesaler can order bakery product.

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Send new order information success and keep into the database. After that the system will redirect to current order page. | Send new order information success and keep into the database. After that the system will redirect to current order page. | Pass |
| 2 | Send new order information unsuccessful and not keep into the database. After that the system gives a wholesaler input new information again. | Send new order information unsuccessful and not keep into the database. After that the system gives a wholesaler input new information again. | Pass |
| 3 | Display error message “The Delivery Date field is required.” After that the system gives a wholesaler input new information again. | Display error message “The Delivery Date field is required.” After that the system gives a wholesaler input new information again. | Pass |
| 4 | Send new order information success and keep into the database. After that the system will redirect to current order page. | Send new order information success and keep into the database. After that the system will redirect to current order page. | Pass |
| 5 | Display unsuccessful message “After midday, you cannot make an order because the bakery product manufacturing in the stage.” | Display unsuccessful message “After midday, you cannot make an order because the bakery product manufacturing in the stage.” | Pass |

**System Test Case-14 (STC-14):** Wholesaler can manage own order information by edit product name, product quantity and delivery date in the order.

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Send new order information success and keep into the database. After that the system will redirect to current order page. | Send new order information success and keep into the database. After that the system will redirect to current order page. | Pass |
| 2 | Send new order information unsuccessful and not keep into the database. After that the system gives a wholesaler input new information again. | Send new order information unsuccessful and not keep into the database. After that the system gives a wholesaler input new information again. | Pass |
| 3 | Display error message “The Delivery Date field is required.” After that the system gives a wholesaler input new information again. | Display error message “The Delivery Date field is required.” After that the system gives a wholesaler input new information again. | Pass |
| 4 | Send new order information success and keep into the database. After that the system will redirect to current order page. | Send new order information success and keep into the database. After that the system will redirect to current order page. | Pass |
| 5 | Display unsuccessful message “After midday, you cannot make an order because the bakery product manufacturing in the stage.” | Display unsuccessful message “After midday, you cannot make an order because the bakery product manufacturing in the stage.” | Pass |

**System Test Case-15 (STC-15):** Wholesaler can manage own order information by cancel the order information from the system.

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Not found order information remain in the system and the system redirect to the order page. | Not found order information remain in the system and the system redirect to the order page. | Pass |
| 2 | Display unsuccessful message “After midday, you cannot cancel the order because the bakery product manufacturing in the stage.” | Display unsuccessful message “After midday, you cannot cancel the order because the bakery product manufacturing in the stage.” | Pass |

**System Test Case-16 (STC-16):** Wholesaler can view their current order’s information.

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Display order information from the system in the current order page. | Display order information from the system in the current order page. | Pass |

**System Test Case-17 (STC-17):** Worker can view current order’s information of all members by search the filter keyword.

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Found information and display order information are related to the filter keyword. | Found information and display order information are related to the filter keyword. | Pass |
| 2 | Not found order information and display the not found message. | Not found order information and display the not found message. | Pass |

**System Test Case-18 (STC-18):** Worker can view total ordered information.

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Display total ordered information from the system in the total ordered page. | Display total ordered information from the system in the total ordered page. | Pass |
| 2 | Found information and display order information are related to the filter keyword. | Found information and display order information are related to the filter keyword. | Pass |
| 3 | Not found order information and display the not found message. | Not found order information and display the not found message. | Pass |

**System Test Case-19 (STC-19):** Admin can mark progressive of order status while the order in the preparation stage.

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Order status has changed and keep in the database. After that the system will display update order status to the web application. | Order status has changed and keep in the database. After that the system will display update order status to the web application. | Pass |

**System Test Case-20 (STC-20):** Wholesaler can view their ordered history.

**Test date:** 24 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Display the ordered history information on the order page. | Display the ordered history information on the order page. | Pass |
| 2 | Found information and display ordered history information are related to the filter keyword. | Found information and display ordered history information are related to the filter keyword. | Pass |
| 3 | Not found ordered history information and display the not found message. | Not found ordered history information and display the not found message. | Pass |

**System Test Case-21 (STC-21):** Worker can view all ordered history.

**Test date:** 22 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Display the ordered history information on the order page. | Display the ordered history information on the order page. | Pass |
| 2 | Found information and display ordered history information are related to the filter keyword. | Found information and display ordered history information are related to the filter keyword. | Pass |
| 3 | Not found ordered history information and display the not found message. | Not found ordered history information and display the not found message. | Pass |

**System Test Case-22 (STC-22):** Admin can manage all ingredient information by create new ingredient information.

**Test date:** 22 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Add ingredient information success and keep in the database. After that the system will redirect to the ingredient management page. | Add ingredient information success and keep in the database. After that the system will redirect to the ingredient management page. | Pass |
| 2 | Display error message “Ingredient Name is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Name is required.” After that the system gives an admin input new information again. | Pass |
| 3 | Display error message “Category is required.” After that the system gives an admin input new information again. | Display error message “Category is required.” After that the system gives an admin input new information again. | Pass |
| 4 | Display error message “Ingredient Picture is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Picture is required.” After that the system gives an admin input new information again. | Pass |
| 5 | Add ingredient information success and keep in the database. After that the system will redirect to the ingredient management page. | Add ingredient information success and keep in the database. After that the system will redirect to the ingredient management page. | Pass |
| 6 | Display error message “Please input with alphabet only.” After that the system gives an admin input new information again. | Display error message “Please input with alphabet only.” After that the system gives an admin input new information again. | Pass |

**System Test Case-23 (STC-23):** Admin can manage all ingredient information by edit the existing ingredient information.

**Test date:** 22 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Add ingredient information success and keep in the database. After that the system will redirect to the ingredient management page. | Add ingredient information success and keep in the database. After that the system will redirect to the ingredient management page. | Pass |
| 2 | Display error message “Ingredient Name is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Name is required.” After that the system gives an admin input new information again. | Pass |
| 3 | Display error message “Category is required.” After that the system gives an admin input new information again. | Display error message “Category is required.” After that the system gives an admin input new information again. | Pass |
| 4 | Display error message “Ingredient Picture is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Picture is required.” After that the system gives an admin input new information again. | Pass |
| 5 | Display error message “Please input with alphabet only.” After that the system gives an admin input new information again. | Display error message “Please input with alphabet only.” After that the system gives an admin input new information again. | Pass |

**System Test Case-24 (STC-24):** Admin can manage all ingredient information by delete the existing ingredient information.

**Test date:** 22 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Ingredient information has deleted and keep updated in the database. After that the system will redirect to the ingredient management page. | Ingredient information has deleted and keep updated in the database. After that the system will redirect to the ingredient management page. | Pass |

**System Test Case-25 (STC-25):** Admin can view all ingredient information by search the filter keywords of ingredient information.

**Test date:** 22 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual Result** | **Pass/Fail** |
| 1 | Display all ingredient list on the ingredient management page. | Display all ingredient list on the ingredient management page. | Pass |
| 2 | Found information and display ingredient information that matched to the filter keyword. | Found information and display ingredient information that matched to the filter keyword. | Pass |
| 3 | Not found ingredient information and display the not found message in the ingredient page. | Not found ingredient information and display the not found message in the ingredient page. | Pass |

**System Test Case-26 (STC-26):** Admin can manage all menu recipe information by create new menu recipe information.

**Test date:** 22 April 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Add menu recipe information success and keep in the database. After that the system will redirect to the recipe management page. | Add menu recipe information success and keep in the database. After that the system will redirect to the recipe management page. | Pass |
| 2 | Display error message “Product Name is required.” After that the system gives an admin input new information again. | Display error message “Product Name is required.” After that the system gives an admin input new information again. | Pass |
| 3 | Display error message “Ingredient Name is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Name is required.” After that the system gives an admin input new information again. | Pass |
| 4 | Display error message “Ingredient Quantity is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Quantity is required.” After that the system gives an admin input new informationgain. | Pass |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 5 | Display error message “Ingredient Units is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Units is required.” After that the system gives an admin input new information again. | Pass |
| 6 | Display error message “Please input with number only.” After that the system gives an admin input new information again. | Display error message “Please input with number only.” After that the system gives an admin input new information again. | Pass |

**System Test Case-27 (STC-27):** Admin can manage all menu recipe information by edit the existing menu recipe information.

**Test date:** 22 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual Result** | **Pass/Fail** |
| 1 | Add menu recipe information success and keep in the database. After that the system will redirect to the recipe management page. | Add menu recipe information success and keep in the database. After that the system will redirect to the recipe management page. | Pass |
| 2 | Display error message “Product Name is required.” After that the system gives an admin input new information again. | Display error message “Product Name is required.” After that the system gives an admin input new information again. | Pass |
| 3 | Display error message “Ingredient Name is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Name is required.” After that the system gives an admin input new information again. | Pass |
| 4 | Display error message “Ingredient Quantity is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Quantity is required.” After that the system gives an admin input new information again. | Pass |
| 5 | Display error message “Ingredient Units is required.” After that the system gives an admin input new information again. | Display error message “Ingredient Units is required.” After that the system gives an admin input new information again. | Pass |
| 6 | Display error message “Please input with number only.” After that the system gives an admin input new information again. | Display error message “Please input with number only.” After that the system gives an admin input new information again. | Pass |

**System Test Case-28 (STC-28):** Admin can manage all menu recipe information by delete the existing menu recipe information.

**Test date:** 22 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Menu recipe information has deleted and keep updated in the database. After that the system will redirect to the recipe management page. | Menu recipe information has deleted and keep updated in the database. After that the system will redirect to the recipe management page. | Pass |

**System Test Case-29 (STC-29):** Admin can view all menu recipe information by search the filter keywords of menu recipe information.

**Test date:** 22 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual Result** | **Pass/Fail** |
| 1 | Display all menu recipe list on the recipe management page. | Display all menu recipe list on the recipe management page. | Pass |
| 2 | Found information and display menu recipe information that matched to the filter keyword. | Found information and display menu recipe information that matched to the filter keyword. | Pass |
| 3 | Not found menu recipe information and display the not found message on the recipe page. | Not found menu recipe information and display the not found message on the recipe page. | Pass |

**System Test Case-30 (STC-30):** Worker can view current ingredient estimation information in the system

**Test date:** 22 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Display current ingredient estimation information from the system in the ingredient estimation page. | Display current ingredient estimation information from the system in the ingredient estimation page. | Pass |

**System Test Case-31 (STC-31):** Worker can view all ingredient estimation history.

**Test date:** 22 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual Result** | **Pass/Fail** |
| 1 | Display the ingredient estimation history information on the ingredient estimation page. | Display the ingredient estimation history information on the ingredient estimation page. | Pass |
| 2 | Found information and display ingredient estimation history information are related to the filter keyword. | Found information and display ingredient estimation history information are related to the filter keyword. | Pass |
| 3 | Not found ingredient estimation history information and display the not found message. | Not found ingredient estimation history information and display the not found message. | Pass |

**System Test Case-32 (STC-32):** Worker and Wholesaler can make a report

**Test date:** 22 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1 | Display the report contents on the report page. | Display the report contents on the report page. | Pass |
| 2 | Export the report contents as PDF file format. | Export the report contents as PDF file format. | Pass |
| 3 | Print out the report contents as a paper document. | Print out the report contents as a paper document. | Pass |

**System Test Case-33 (STC-33):** Worker and Wholesaler can make a graph

**Test date:** 22 June 2015

**Test by:** Parinya Panyanak

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Expected output** | **Actual Result** | **Pass/Fail** |
| 1 | Display the graph image on the graph page. | Display the graph image on the graph page. | Pass |
| 2 | Export the graph image as PDF file format. | Export the graph image as PDF file format. | Pass |
| 3 | Print out the graph image as a paper document. | Print out the graph image as a paper document. | Pass |